

2F00189-US-P

Japanese Laid-Open Patent Publication No. S63-310029 (line 14,
lower right column of page 3 through line 7, upper left column
5 of page 4)

The interrupt processing system according to the present
invention is a system in which a CPU is able to perform multiplex
interrupt processing in accordance with interrupt request
10 signals with a plurality of interrupt causes, where the system
is provided with two interrupt acceptance flags for determining
whether to carry out the interruption or not in accordance with
the aforementioned respective interrupt causes, one of the
aforementioned interrupt acceptance flag is adapted to instruct
15 the interruption in accordance with the predetermined priority
order for each of the aforementioned interrupt causes.

[Operation]

Because one of the interrupt acceptance flags according
20 to the present invention instructs the interruption in
accordance with the predetermined priority order for each of
interrupt causes, it is possible to realize multiplex interrupt
processing by saving, setting, and restoring the interrupt
acceptance flag in batch.

⑫ 公開特許公報(A)

昭63-310029

⑮ Int. Cl.⁴

識別記号

庁内整理番号

⑬ 公開 昭和63年(1988)12月19日

G 06 F 9/46

3 1 1

C-7056-5B

審査請求 未請求 発明の数 1 (全11頁)

⑭ 発明の名称 割込み処理システム

⑯ 特 願 昭62-147235

⑰ 出 願 昭62(1987)6月11日

⑱ 発 明 者 竹 内 稔 兵庫県伊丹市瑞原4丁目1番地 三菱電機株式会社北伊丹製作所内

⑲ 出 願 人 三菱電機株式会社 東京都千代田区丸の内2丁目2番3号

⑳ 代 理 人 弁理士 大岩 増雄 外2名

明 細 書

1. 発明の名称

割込み処理システム

2. 特許請求の範囲

(1) CPUが複数の割込み要因の割込み要求信号に従い、多重割込み処理が行える割込み処理システムにおいて、

前記割込み要因にそれぞれ対応させて、その割込みを実行するか否かを決定する割込み許可フラグを2つ設け、

前記割込み許可フラグの一方は、前記割込み要因各々の予め定められた優先順位に従った割込みを指示するものであることを特徴とする割込み処理システム。

3. 発明の詳細な説明

(産業上の利用分野)

この発明は多重割込み処理が行える割込み処理システムに関し、特にマイクロプロセッサに接続または内蔵されることで多重割込みを実現する割込み処理システムに関する。

(従来の技術)

従来のマイクロプロセッサで複数の割込み要因に対する割込み処理を行なう多重割込み処理システムとして、例えば第3図のシステム構成図に示したものがあつた。同図において、1はCPUであり、割込みマスクフラグ(以下「1フラグ」と言う。)1aを備えている。また2はメモリ、3はバスである。R(R_a~R_d)はレジスタであり、各割込み要因A~Dに対応して4つ設けられており、このレジスタRは割込み要求フラグF₁(F_{1a}~F_{1d})と割込み許可フラグF₂(F_{2a}~F_{2d})を持っている。なお他の機能についてはこの発明とは関係ないので言及しない。割込み要求フラグF₁の入力部は各割込み要因A~Dが発生すると、“1”をセットする(通常は“0”)ように各割込み要因A~Dと接続され、このフラグF₁の出力部は2入力ANDゲートG(G_a~G_d)の一方入力となつてゐる。また、割込み許可フラグF₂はANDゲートGの他方入力となつてゐる。なお、これらのフラグF₁、F₂はバス

3を介して、CPU1によりアクセス可能である。また、ANDゲートGの出力信号 S_I ($S_{1a} \sim S_{1d}$)がCPU1への各割込み要因A～Dによる割込み要求信号となっている。

このような構成において、例えば割込み要因Aが発生した場合を考える。この場合、まず割込み要求フラグ F_{1a} が“1”にセットされる。この時、割込み許可フラグ F_{2a} が“1”であれば、ANDゲート G_a の出力信号、つまり割込み要求信号 S_{1a} が“1”となることで、割込み要求信号 S_{1a} がイネーブル状態となる。ここでCPU1のIフラグ $1a$ が“0”であれば、CPU1は割込み受付け状態であるので、CPU1は現在実行中の処理を中断し、プログラムカウンタ、プログラム状態語 (PSW)、各種レジスタ値等の退避の割込み受付け処理を行ない、割込み要因Aに基づく割込み処理を行う。しかる後、当該割込み処理が終了すれば、割込み受付け時に退避したプログラムカウンタ、PSW、各種レジスタ値等の復帰の割込み終了処理を施した後、割込み要因Aにより中

断された状態から処理が再開される。

一方、割込み要求Aが発生し、割込み要求フラグ F_{1a} が“1”にセットされても、割込み許可フラグ F_{2a} が“0”であれば、割込み要求信号 S_{1a} が“0”で、イネーブル状態とならないため、割込み要求Aは受付けられない。

また、割込み要求フラグ F_{1a} 、割込み許可フラグ F_{2a} が共に“1”となり、割込み要求信号 S_{1a} がイネーブル状態となっても、CPU1のIフラグ $1a$ が“1”の時は、CPU1はこの割込みを受付けず、Iフラグ $1a$ が“0”になるまで、その割込み要求は保留される。上記した一連の処理は割込み要因B～Dにおいても同様である。

上記したシステムでは割込み要求信号 S_I がイネーブル状態となりIフラグ $1a$ が“0”の状態の時、CPU1で割込み処理が受付けられると、他の割込み要求を保留させるため、Iフラグ $1a$ は必ず“1”に設定されるようにしている。このため1つの割込み処理実行中に、この割込み処理より優先順位の高い割込み処理を行うことのでき

る、いわゆる多重割込み処理を実現するためには、以下に示すようなソフトウェア処理が必要となる。

第4図はそのソフトウェア処理の処理手順を示すフローチャートであり、割込み優先順位は割込み要因A、B、C、Dの順とする。同図において「要因A (B、C、D)の割込み」とは、各割込み要因A～Dによる割込み要求信号 S_I がイネーブル状態で、かつIフラグ $1a$ が“0”でCPU1が当該割込み要求を受付けたことを示し、受付けると同時にIフラグ $1a$ を“1”と設定し、他の割込みを受付けなくした処理を示している。

そして、「要因A (B、C、D)に対する処理」とは、前述したプログラムカウンタ、PSW、各種レジスタ値等の退避、復帰等の割込み受付け処理、割込み終了処理を含んだ割込み要因A (B、C、D)に基づく割込み処理を示している。また、「割込み終了」時に他の割込みを受付けるためIフラグ $1a$ を“0”にクリアしており、割込み許可フラグ $F_{2a} \sim F_{2d}$ の初期設定値は全て“1”であるとする。

以下、同図を参照しつつこの動作の説明をする。まず、割込み要因Aの割込みが実行される場合について述べる。この場合、無条件にステップS1で割込み要因Aに対する処理を行い終了する。この間Iフラグ $1a$ は“1”であるので、割込み要因Aを含む全ての割込み要求信号 S_I が保留されることになるので、他の割込み処理が実行されることはない。

次に、割込み要因Bの割込みが実行される場合について述べる。この場合、ステップS11で、割込み許可フラグ F_{2b} を“0”にクリアすること、割込み要求信号 S_{1b} が決してイネーブル状態にならないようにする。次にステップS12で割込み許可フラグ F_{2c} の値を退避させ、ステップS13で割込み許可フラグ F_{2c} を“0”にクリアする。さらに、ステップS14で割込み許可フラグ F_{2d} の値を退避させ、ステップS15で割込み許可フラグ F_{2d} を“0”にクリアする。ステップS12～S15の処理は、割込み要因Bより優先順位の低い割込み要因C、Dによる割込みを禁止し、

割込み要因Bの割込み処理終了後は割込み許可フラグF_{2c}・F_{2d}を元の値に戻すために行っている。

そして、ステップS16においてIフラグ1aの値を強制的に“0”にクリアし、CPU1自体は割込み受付状態とし、次のステップS17で割込み要因Bに対する処理を行う。このステップS17の実行中、割込み許可フラグF_{2a}～F_{2d}は各々“1”、“0”、“0”、“0”でIフラグ1aは“0”であることより、割込み要因Bより優先順位の高い割込み要因Aのみ割込み可能としている。

以下、ステップS18でIフラグ1aを“1”にセットすることで、全ての割込み要求信号S₁を保留させ、ステップS19で割込み許可フラグF_{2d}の値を復帰させ、ステップS20で割込み許可フラグF_{2c}の値を復帰させ、割込み許可フラグF_{2c}・F_{2d}を割込み要因Bの割込み直前の状態に戻し、ステップS21で割込み許可フラグF_{2b}を“1”にセットし、割込み要因Bの割込みを可能にし、終了する。

割込み要因C、Dによる割込みにおいても、優先順位の違いを除き、その処理手順はそれぞれステップS31～S38およびステップS41～S45に示すように割込み要因Bの場合と同じである。参考までに割込み要因A～Dに対する処理中（ステップS1、S17、S35、S43実行中）における割込み許可フラグF_{2a}～F_{2d}とIフラグ1aの値を表1に示す。

(以下余白)

表 1

処理	Iフラグ	割込み許可フラグ			
		F _{2a}	F _{2b}	F _{2c}	F _{2d}
割込み要因A	1	1	Δ	Δ	Δ
割込み要因B	0	1	0	0	0
割込み要因C	0	1	1	0	0
割込み要因D	0	1	1	1	0

Δ：割込み要因Aが受け付けられる直前の値

なお、上記した処理は割込み許可フラグF_{2a}～F_{2d}の初期設定値は全て“1”である場合について述べたが、複数を含むいずれかの初期設定値を“0”にすることで、当該割込み要因を優先順位に関係なく受け付けなくすることができる。

(発明が解決しようとする問題点)

従来の多重割込み処理を行う割込み処理システムは、以上のように構成されているので、第4図で示したような多重割込み処理実現のためのソフトウェア処理を必要とし、この処理量は割込み要因の数が増えるに従い膨大になってしまい、割込み処理システムの処理能力低下が生じてしまうという問題点があった。

この発明は上記のような問題点を解消するためになされたもので、多重割込み処理実現のための処理量を低減させ、かつシステムとしての処理能力の低下をおこさない割込み処理システムを得ることを目的とする。

(問題点を解決するための手段)

この発明にかかる割込み処理システムは、CPUが複数の割込み要因の割込み要求信号に従い、多重割込み処理が行えるシステムであって、前記割込み要因にそれぞれ対応させて、その割込みを実行するか否かを決定する割込み許可フラグを2つ設け、前記割込み許可フラグの一方は、前記割込み要因各々の予め定められた優先順位に従った

割込みを指示するようにしている。

(作用)

この発明における一方の割込み許可フラグは割込み要因の各々の予め定められた優先順位に従った割込みを指示するため、この割込み許可フラグを一括して退避、セット、復帰することで多重割込み処理を実現することができる。

(実施例)

第1図はこの発明の一実施例である多重割込み処理を行う割込み処理システムを示すシステム構成図である。同図において1、2、R、F₁、F₂、S₁は従来と同じなので説明を省略するが、新たに割込み許可フラグF₃(F_{3a}~F_{3d})を有する許可レジスタPRが設けられた点異なる。この許可レジスタPRはバス3を介してCPU1によりアクセス可能にしてあり、各割込み許可フラグF_{3a}~F_{3d}はANDゲートG_a~G_dの入力部に接続されている。従って、ANDゲートGは割込み要求フラグF₁、割込み許可フラグF₂、F₃の3入力となる。

S54の実行中、割込み許可フラグF_{2a}~F_{2d}が全て“1”、Iフラグ1aが“0”であるが、割込み許可フラグF_{3a}~F_{3d}は全て“0”であることより、全ての割込みを禁止している。

以下、ステップS55でIフラグ1aを“1”にセットし、全ての割込み要求信号S₁を保留させ、ステップS56で割込み許可フラグF_{3a}~F_{3d}の値を復帰させ、割込み要因Aの割込み直前の状態に戻し、割込み処理を終了する。

次に割込み要因Bの割込みが実行される場合について述べる。この場合、ステップS61で許可レジスタPR内の割込み許可フラグF_{3a}~F_{3d}の値を退避させ、次のステップS62で割込み許可フラグF_{3a}、F_{3b}、F_{3c}、F_{3d}を“1”、“0”、“0”、“0”にセットする。このステップS61、S62における処理は割込み要因Bより優先順位の低い割込み(割込み要因B、C、D)を禁止し、割込み要因Bより高い優先順位の割込み(割込み要因A)を受付けるようにし、さらに割込み許可フラグF_{3a}~F_{3d}の値を割込み要因B

第2図は第1図で示した割込み処理システムにおける多重割込み処理をソフトウェアで実現するための処理手順を示すフローチャートである。ここでレジスタR内の各割込み許可フラグF₂及び許可レジスタPR内の各許可フラグF₃は“1”に設定しておく。以下同図を参照しつつその動作の説明をする。

まず、割込み要因Aの割込みが実行される場合について述べる。この場合、ステップS51で許可レジスタPR内の割込み許可フラグF_{3a}~F_{3d}の値を全て退避させる。次にステップS52で割込み許可フラグF_{3a}~F_{3d}の値を全て“0”にクリアする。このステップS51、S52における処理は全ての割込みを禁止し、割込み許可フラグF_{3a}~F_{3d}を割込み要因Aの割込み処理終了後に元の状態に戻すために行っている。

そして、ステップS53においてIフラグ1aの値を強制的に“0”にクリアし、CPU1自体は割込み受け状態とし、次のステップS54で割込み要因Aに対する処理を行う。このステップ

の割込み処理終了後に元に戻すために行っている。

そして、ステップS63においてIフラグ1aの値を強制的に“0”にクリアし、CPU1自体は割込み受け状態とし、次のステップS64で割込み要因Bに対する処理を行う。このステップS64の実行中、割込み許可フラグF_{2a}~F_{2d}は全て“1”、割込み許可フラグF_{3a}~F_{3d}は各々“1”、“0”、“0”、“0”でIフラグ1aは“0”であることにより、割込み要因Bより優先順位の低い割込み要因Aのみ割込み可能にしている。

以下、ステップS65でIフラグ1aを“1”にセットすることで全ての割込み要求信号S₁を保留させ、ステップS66で割込み許可フラグF_{3a}~F_{3d}の値を復帰させ、割込み要因Bの割込み直前の状態に戻し、割込み処理を終了する。

割込み要因C、Dによる割込みにおいても、優先順位の違いを除き、その処理手順はそれぞれステップS71~S76およびステップS81~S

86に示すように割込み要因A、Bの場合と同じである。

上記したように、許可レジスタPRを新たに設けたことにより、多重割込み処理を行なうときには許可レジスタPRの内容を一括して退避、再設定、復帰すればよく、従来のように1割込み要因ごとに割込み許可フラグ F_2 の退避、再設定、復帰を行う必要はないため、多重割込み処理実現のソフトウェア処理が軽減される。すなわち、割込み要因の数が増えてもその処理は、許可レジスタPR内に割込み許可フラグ F_3 が設定できる範囲では、割込み個々に対する処理は、割込み許可フラグ F_3 の設定以外は全く同一である。従って、割込み許可フラグ F_3 の設定は分岐命令等により行うことで1つのプログラムルーチンで多重割込み処理が実現できる。その結果、割込み要因数が増えても割込み処理システムの処理能力低下は全く生じない。

なお、第2図で示したソフトウェアによる多重割込み処理は、許可レジスタPR内の割込み許可

フラグ F_3 の退避、再設定、復帰という単純な処理であるので、CPU1の割込み受け付け動作及び割込み完了動作としてハードウェア処理で行うことも十分可能である。この場合、割込み許可フラグ F_3 の再設定に使用する値を各割込み要因ごとにメモリの予め定められたアドレスに格納しておき、再設定を行なうときには該当のアドレスからその値を読出して許可レジスタPRに転送するようにしておくことで、該アドレスに格納する値を変更することにより優先順位を任意に変更できる。

また、この実施例では割込み許可フラグ $F_{2a} \sim F_{2d}$ の初期設定値は全て“1”である場合について述べたが、複数を含むいずれかの初期設定値を“0.”にすることで許可レジスタPR内の割込み許可フラグ F_3 の値で決定する優先順位に関係なく、当該割込み要因を受け付けなくすることもできる。

(発明の効果)

以上説明したように、この発明によれば、一方の割込み許可フラグの値により割込み要因各々の

優先順位に従った割込みを指示するため、多重割込み実現のための処理量を低減させ、かつシステムとしての処理能力の低下をおこさない割込み処理システムを得ることができる。

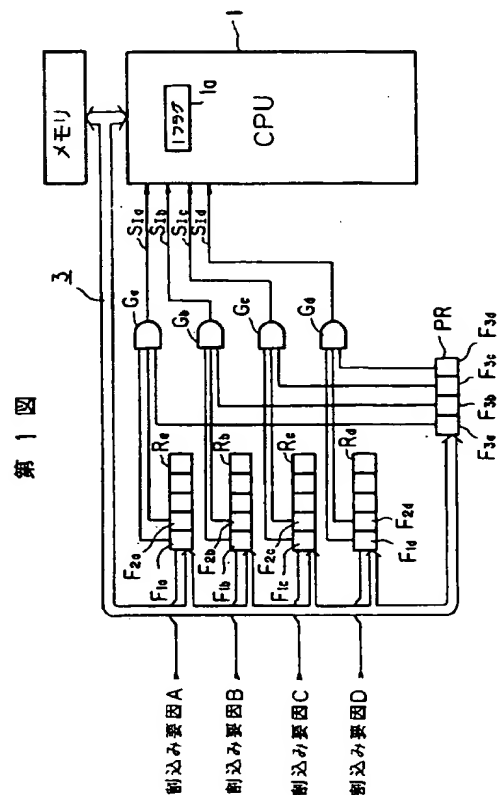
4. 図面の簡単な説明

第1図はこの発明の一実施例である割込み処理システムを示すシステム構成図、第2図は第1図の割込み処理システムを用いて多重割込み処理を実現するための処理手順を示すフローチャート、第3図は従来の割込み処理システムを示すシステム構成図、第4図は第3図の割込み処理システムを用いて多重割込み処理を実現するための処理手順を示すフローチャートである。

図において、1はCPU、3はバス、PRは許可レジスタ、 F_2 、 F_3 は割込み許可フラグである。

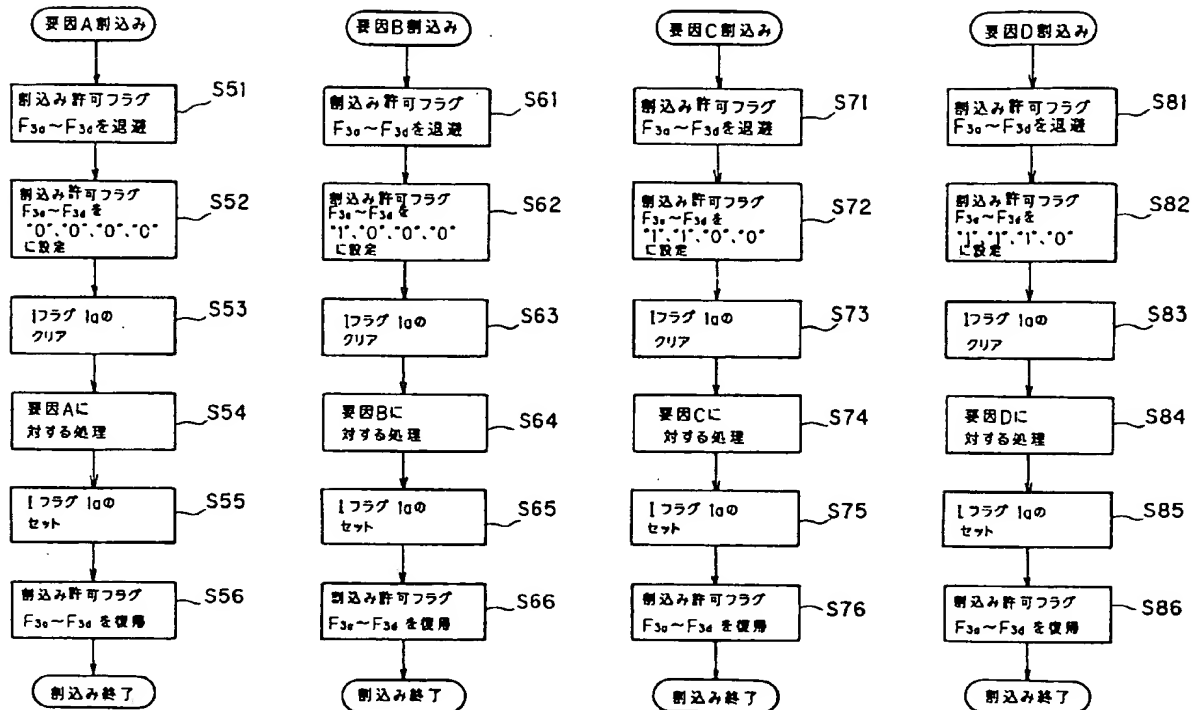
なお、各図中同一符号は同一または相当部分を示す。

代理人 大 岩 増 雄

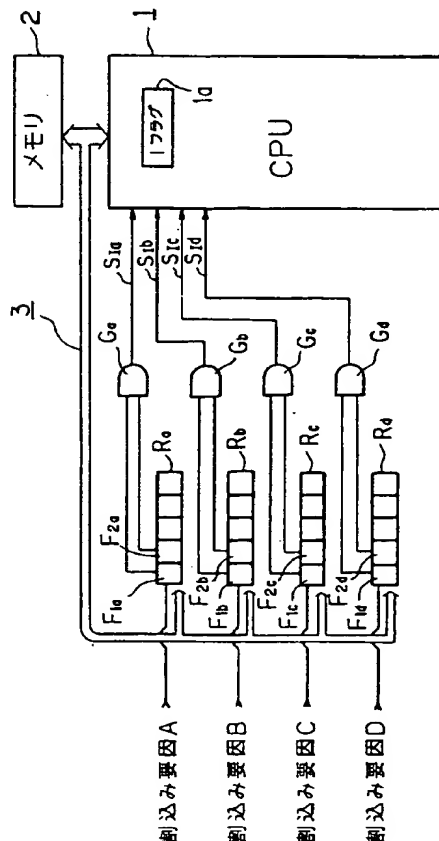


1...CPU
3...バス
PR...許可レジスタ
 F_2, F_3 ...割込み許可フラグ

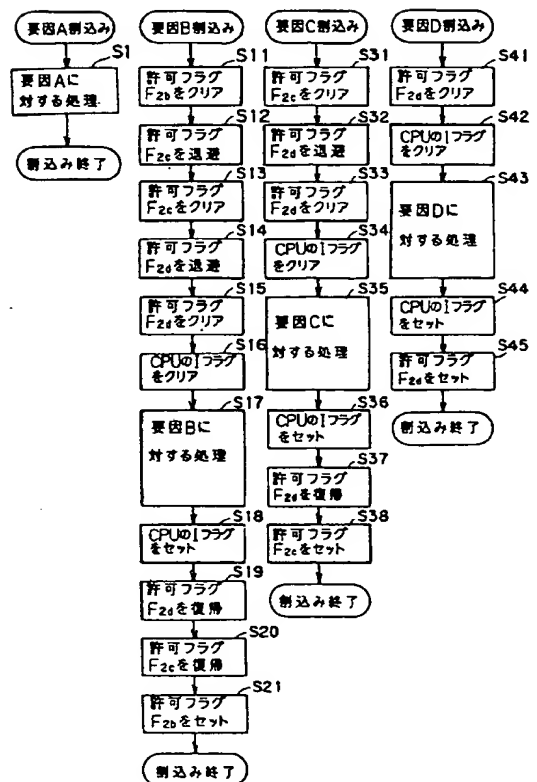
第 2 図



第 3 図



第 4 図



手続補正書(自発)
63 1 8
昭和 年 月 日

特許庁長官殿

1. 事件の表示 特願昭62-147235号

2. 発明の名称

割込み処理システム

3. 補正をする者

事件との関係 特許出願人
住 所 東京都千代田区丸の内二丁目2番3号
名 称 (601)三菱電機株式会社
代表者 志 岐 守 哉

4. 代 理 人

住 所 東京都千代田区丸の内二丁目2番3号
三菱電機株式会社内
氏 名 (7375)弁理士 大 岩 増 雄
(連絡先03(213)3421特許部)

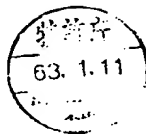
5. 補正の対象

明細書の「特許請求の範囲の欄」及び「発明の詳細な説明の欄」

6. 補正の内容

(1) 明細書の「特許請求の範囲の欄」、「発明の詳細な説明の欄」を別紙の通り(補正の対象の欄に記載した事項以外は内容に変更なし)補正する。

以上



明 細 書

1. 発明の名称

割込み処理システム

2. 特許請求の範囲

(1) CPUが複数の割込み要因の割込み要求信号に従い、多重割込み処理が行える割込み処理システムにおいて、

前記割込み要因にそれぞれ対応させて、その割込みを実行するか否かを決定する割込み許可フラグを2つ設け、

前記割込み許可フラグの一方は、前記割込み要因各々の優先順位に従った割込み許可/禁止状態を指示するものであることを特徴とする割込み処理システム。

3. 発明の詳細な説明

(産業上の利用分野)

この発明は多重割込み処理が行える割込み処理システムに関し、特にマイクロプロセッサに接続または内蔵されることで多重割込みを実現する割込み処理システムに関する。

(従来の技術)

従来のマイクロプロセッサで複数の割込み要因に対する割込み処理を行なう多重割込み処理システムとして、例えば第3図のシステム構成図に示したものがあつた。同図において、1はCPUであり、割込みマスクフラグ(以下「Iフラグ」と言う。)1aを備えている。また2はメモリ、3はバスである。R(R_a~R_d)はレジスタであり、各割込み要因A~Dに対応して4つ設けられており、このレジスタRは割込み要求フラグF₁(F_{1a}~F_{1d})と割込み許可フラグF₂(F_{2a}~F_{2d})を持っている。なお他の機能についてはこの発明とは関係ないので言及しない。割込み要求フラグF₁の入力部は各割込み要因A~Dが発生すると、“1”をセットする(通常は“0”)ように各割込み要因A~Dと接続され、このフラグF₁の出力部は2入力ANDゲートG(G_a~G_d)の一方入力となっている。また、割込み許可フラグF₂はANDゲートGの他方入力となっている。なお、これらのフラグF₁、F₂はバス

3を介して、CPU1によりアクセス可能である。また、ANDゲートGの出力信号 S_1 ($S_{1a} \sim S_{1d}$)がCPU1への各割込み要因A～Dによる割込み要求信号となっている。

このような構成において、例えば割込み要因Aが発生した場合を考える。この場合、まず割込み要求フラグ F_{1a} が“1”にセットされる。この時、割込み許可フラグ F_{2a} が“1”であれば、ANDゲート G_a の出力信号、つまり割込み要求信号 S_{1a} が“1”となることで、割込み要求信号 S_{1a} がイネーブル状態となる。ここでCPU1のIフラグ $1a$ が“0”であれば、CPU1は割込み受け付け許可状態であるので、CPU1は現在実行中の処理を中断し、プログラムカウンタ、プログラム状態語(PSW)、各種レジスタ値等の退避の割込み受け付け処理を行ない、割込み要因Aに基づく割込み処理を行う。しかる後、当該割込み処理が終了すれば、割込み受け付け時に退避したプログラムカウンタ、PSW、各種レジスタ値等の復帰の割込み終了処理を施し、割込み要因Aにより中

断された状態から処理が再開される。

一方、割込み要求Aが発生し、割込み要求フラグ F_{1a} が“1”にセットされても、割込み許可フラグ F_{2a} が“0”であれば、割込み要求信号 S_{1a} が“0”で、イネーブル状態とならないため、割込み要求Aは受け付けられない。

また、割込み要求フラグ F_{1a} 、割込み許可フラグ F_{2a} が共に“1”となり、割込み要求信号 S_{1a} がイネーブル状態となっても、CPU1のIフラグ $1a$ が“1”の時は、CPU1はこの割込みを受け付けず、Iフラグ $1a$ が“0”になるまで、その割込み要求は保留される。上記した一連の処理は割込み要因B～Dにおいても同様である。

上記したシステムでは割込み要求信号 S_1 がイネーブル状態となりIフラグ $1a$ が“0”の状態の時、CPU1で割込み処理が受け付けられると、他の割込み要求を保留させるため、Iフラグ $1a$ は必ず“1”に設定されるようにしている。このため1つの割込み処理実行中に、この割込み処理より優先順位の高い割込み処理を行うことのでき

る、いわゆる多重割込み処理を実現するためには、以下に示すようなソフトウェア処理が必要となる。

第4図はそのソフトウェア処理の処理手順を示すフローチャートであり、割込み優先順位は割込み要因A、B、C、Dの順とする。同図において「要因A(B、C、D)の割込み」とは、各割込み要因A～Dによる割込み要求信号 S_1 がイネーブル状態で、かつIフラグ $1a$ が“0”でCPU1が当該割込み要求を受け付けたことを示し、受け付けると同時に前述したプログラムカウンタ、PSW、各種レジスタの退避等を行なった後、Iフラグ $1a$ を“1”と設定し、他の割込みを受け付けなくした処理を示している。

そして、「要因A(B、C、D)に対する処理」とは、割込み要因A(B、C、D)の割込み要求に基づく割込み処理を示している。また、「割込み終了」とは他の割込みを受け付けるためIフラグ $1a$ を“0”にクリアした後、割込受け付け時に退避した各種レジスタ、PSW、プログラムカウンタを復帰し、割込受け付けにより中断された状態から処

理を再開することを示している。

以下、同図を参照しつつこの動作の説明をする。まず、割込み要因Aの割込みが実行される場合について述べる。この場合、無条件にステップS1で割込み要因Aに対する処理を行い終了する。この間Iフラグ $1a$ は“1”であるので、割込み要因Aを含む全ての割込み要求信号 S_1 が保留されることになるので、他の割込み処理が実行されることはない。

次に、割込み要因Bの割込みが実行される場合について述べる。この場合、ステップS11で、割込み許可フラグ F_{2b} を“0”にクリアすることで、割込み要求信号 S_{1b} が決してイネーブル状態にならないようにする。次にステップS12で割込み許可フラグ F_{2c} の値を退避させ、ステップS13で割込み許可フラグ F_{2c} を“0”にクリアする。さらに、ステップS14で割込み許可フラグ F_{2d} の値を退避させ、ステップS15で割込み許可フラグ F_{2d} を“0”にクリアする。ステップS12～S15の処理は、割込み要因Bより優先順

位の低い割込み要因C、Dによる割込みを禁止し、割込み要因Bの割込み処理終了後は割込み許可フラグ F_{2c} 、 F_{2d} を元の値に戻すために行っている。

そして、ステップS16においてIフラグ1aの値を“0”にクリアし、CPU1自体は割込み受け付け許可状態とし、次のステップS17で割込み要因Bに対する処理を行う。このステップS17の実行中、割込み許可フラグ F_{2a} は割込み受け付け直前の値であり、 $F_{2b} \sim F_{2d}$ は全て“0”でIフラグ1aは“0”であることより、割込み要因B及び割込み要因Bより優先順位の低い割込み要因を全て割込み禁止としている。

以下、ステップS18でIフラグ1aを“1”にセットすることで、全ての割込み要求信号 S_1 を保留させ、ステップS19で割込み許可フラグ F_{2d} の値を復帰させ、ステップS20で割込み許可フラグ F_{2c} の値を復帰させ、割込み許可フラグ F_{2c} 、 F_{2d} を割込み要因Bの割込み直前の状態に戻し、ステップS21で割込み許可フラグ F_{2b} を“1”にセットし、割込み要因Bの割込みを可能

にし、割込み終了する。

割込み要因C、Dによる割込みにおいても、優先順位の違いを除き、その処理手順はそれぞれステップS31～S38およびステップS41～S45に示すように割込み要因Bの場合と同じである。参考までに割込み要因A～Dに対する処理中（ステップS1、S17、S35、S43実行中）における割込み許可フラグ $F_{2a} \sim F_{2d}$ とIフラグ1aの値を表1に示す。

(以下余白)

表 1

処理	Iフラグ	割込み許可フラグ			
		F_{2a}	F_{2b}	F_{2c}	F_{2d}
割込み 要因 A	1	1	Δ	Δ	Δ
割込み 要因 B	0	Δ	0	0	0
割込み 要因 C	0	Δ	Δ	0	0
割込み 要因 D	0	Δ	Δ	Δ	0

Δ：割込みが受け付けられる直前の値

(発明が解決しようとする問題点)

従来の多重割込み処理を行う割込み処理システムは、以上のように構成されているので、第4図で示したような多重割込み処理実現のためのソフトウェア処理を必要とし、この処理量は割込み要因の数が増えるに従い膨大になってしまい、割込

み処理システムの処理能力低下が生じてしまうという問題点があった。

この発明は上記のような問題点を解消するためになされたもので、多重割込み処理実現のための処理量を低減させ、かつシステムとしての処理能力の低下を最小限に抑える割込み処理システムを得ることを目的とする。

(問題点を解決するための手段)

この発明にかかる割込み処理システムは、CPUが複数の割込み要因の割込み要求信号に従い、多重割込み処理が行えるシステムであって、前記割込み要因にそれぞれ対応させて、その割込みを実行するか否かを決定する割込み許可フラグを2つ設け、前記割込み許可フラグの一方は、前記割込み要因各々の優先順位に従った割込み許可／禁止状態を指示するようにしている。

(作用)

この発明における一方の割込み許可フラグは割込み要因の各々の優先順位に従った割込み許可／禁止状態を指示するため、この割込み許可フラグ

を一括して退避、セット、復帰することで多重割込み処理を実現することができる。

(実施例)

第1図はこの発明の一実施例である多重割込み処理を行う割込み処理システムを示すシステム構成図である。同図において1、2、R、F₁、F₂、S₁は従来と同じなので説明を省略するが、新たに割込み許可フラグF₃(F_{3a}~F_{3d})を有する許可レジスタPRが設けられた点異なる。この許可レジスタPRはバス3を介してCPU1によりアクセス可能にしてあり、各割込み許可フラグF_{3a}~F_{3d}はANDゲートG_a~G_dの入力部に接続されている。従って、ANDゲートGは割込み要求フラグF₁、割込み許可フラグF₂、F₃の3入力となる。

第2図は第1図で示した割込み処理システムにおける多重割込み処理をソフトウェアで実現するための処理手順を示すフローチャートである。ここでレジスタR内の各割込み許可フラグF₂はソフトウェアのフローに従った割込許可/禁止に対

り、全ての割込みを禁止している。

以下、ステップS55でIフラグ1aを“1”にセットし、全ての割込み要求信号S₁を保留させ、ステップS56で割込み許可フラグF_{3a}~F_{3d}の値を復帰させ、割込み要因Aの割込み直前の状態に戻し、割込み処理を終了する。

次に割込み要因Bの割込みが実行される場合について述べる。この場合、ステップS61で許可レジスタPR内の割込み許可フラグF_{3a}~F_{3d}の値を退避させ、次のステップS62で割込み許可フラグF_{3a}、F_{3b}、F_{3c}、F_{3d}を“1”、“0”、“0”、“0”にセットする。このステップS61、S62における処理は割込み要因Bより優先順位の高くはない割込み(割込み要因B、C、D)を禁止し、割込み要因Bより高い優先順位の割込み(割込み要因A)が割込み許可フラグF₂により許可されている場合に受付けるようにし、さらに割込み許可フラグF_{3a}~F_{3d}の値を割込み要因Bの割込み処理終了後に元に戻すために行っている。

応する値に設定し、一方許可レジスタPR内の各許可フラグF₃は“1”に設定しておく。以下同図を参照しつつその動作の説明をする。

まず、割込み要因Aの割込みが実行される場合について述べる。この場合、ステップS51で許可レジスタPR内の割込み許可フラグF_{3a}~F_{3d}の値を全て退避させる。次にステップS52で割込み許可フラグF_{3a}~F_{3d}の値を全て“0”にクリアする。このステップS51、S52における処理は全ての割込みを禁止し、割込み許可フラグF_{3a}~F_{3d}を割込み要因Aの割込み処理終了後に元の状態に戻すために行っている。

そして、ステップS53においてIフラグ1aの値を“0”にクリアし、CPU1自体は割込み受付け状態とし、次のステップS54で割込み要因Aに対する処理を行う。このステップS54の実行中、割込み許可フラグF_{2a}~F_{2d}はソフトウェアのフローにより許可されている要因のものが“1”、Iフラグ1aが“0”であるが、割込み許可フラグF_{3a}~F_{3d}は全て“0”であることよ

そして、ステップS63においてIフラグ1aの値を“0”にクリアし、CPU1自体は割込み受付け状態とし、次のステップS64で割込み要因Bに対する処理を行う。このステップS64の実行中、割込み許可フラグF_{2a}~F_{2d}はソフトウェアのフローにより許可されているものは“1”、割込み許可フラグF_{3a}~F_{3d}は各々“1”、“0”、“0”、“0”でIフラグ1aは“0”であることにより、割込み要因Bより優先順位の低い割込み要因を全て割込み禁止とし、優先順位の高い割込み要因Aは割込み許可フラグF_{2a}がソフトウェアのフローにより“1”がセットされている場合に割込み許可にしている。

以下、ステップS65でIフラグ1aを“1”にセットすることで全ての割込み要求信号S₁を保留させ、ステップS66で割込み許可フラグF_{3a}~F_{3d}の値を復帰させ、割込み要因Bの割込み直前の状態に戻し、割込み処理を終了する。

割込み要因C、Dによる割込みにおいても、優先順位の違いを除き、その処理手順はそれぞれス

テップS71～S76およびステップS81～S86に示すように割込み要因A、Bの場合と同じである。

上記したように、許可レジスタPRを新たに設けたことにより、多重割込み処理を行なうときには許可レジスタPRの内容を一括して退避、再設定、復帰すればよく、従来のように1割込み要因ごとに割込み許可フラグF₂の退避、再設定、復帰を行う必要はないため、多重割込み処理実現のソフトウェア処理が軽減される。すなわち、割込み要因の数が増えても割込み個々に対する処理は、許可レジスタPR内に割込み許可フラグF₃が設定できる範囲では、割込み許可フラグF₃の設定以外は全く同一である。従って、割込み要因数が増えても割込み処理システムの処理能力の低下は最小限に押えることができる。

なお、第2図で示したソフトウェアによる多重割込み処理は、許可レジスタPR内の割込み許可フラグF₃の退避、再設定、復帰という単純な処理であるので、CPU1の割込み受け動作及び

割込み完了動作としてハードウェア処理で行うことも十分可能である。この場合、割込み許可フラグF₃の再設定に使用する値を各割込み要因ごとにメモリの予め定められたアドレスに格納しておき、再設定を行なうときには該当のアドレスからその値を読み出して許可レジスタPRに転送するようにしておくことで、該アドレスに格納する値を変更することにより優先順位を任意に変更できる。

(発明の効果)

以上説明したように、この発明によれば、各割込要因に対して割込み許可フラグを2つ設け、一方を優先順位決定専用で使用し、割込受け及び終了時に優先順位に従い一括して退避、再設定、復帰を行うため、多重割込み実現のための処理量を低減させ、かつシステムとしての処理能力の低下を最小限に押える割込み処理システムを得ることができる。

4. 図面の簡単な説明

第1図はこの発明の一実施例である割込み処理システムを示すシステム構成図、第2図は第1図

の割込み処理システムを用いて多重割込み処理を実現するための処理手順を示すフローチャート、第3図は従来の割込み処理システムを示すシステム構成図、第4図は第3図の割込み処理システムを用いて多重割込み処理を実現するための処理手順を示すフローチャートである。

図において、1はCPU、3はバス、PRは許可レジスタ、F₂、F₃は割込み許可フラグである。

なお、各図中同一符号は同一または相当部分を示す。

代理人 大 岩 増 雄